2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

A Real-time Pedestrian Detection System based on Structure and Appearance Classification

Mayank Bansal, Sang-Hack Jung, Bogdan Matei, Jayan Eledath, Harpreet Sawhney
Vision Technologies, Sarnoff Corporation
201 Washington Road, Princeton, NJ 08540, USA
{mbansal,sjung,bmatei,jeledath,hsawhney}@sarnoff.com

***Abstract*— We present a real-time pedestrian detection system based on structure and appearance classification. We discuss several novel ideas that contribute to having low-false alarms and high detection rates, while at the same time achieving computational efficiency: (i) At the front end of our system we employ stereo to detect pedestrians in 3D range maps using template matching with a representative 3D shape model, and to classify other background objects in the scene such as buildings, trees and poles. The structure classification efficiently labels substantial amount of non-relevant image regions and guides the further computationally expensive process to focus on relatively small image parts; (ii) We improve the appearance-based classifiers based on HoG descriptors by performing template matching with 2D human shape contour fragments that results in improved localization and accuracy; (iii) We build a suite of classifiers tuned to specific distance ranges for optimized system performance. Our method is evaluated on publicly available datasets and is shown to match or exceed the performance of leading pedestrian detectors in terms of accuracy as well as achieving real-time computation (10 Hz), which makes it adequate for in-vehicle navigation platform.**

I. Introduction

This paper describes a real time stereo-based pedestrian detection system for autonomous robot navigation, which achieves a low false alarm rate per frame at above 90% detection accuracy. Cameras provide a passive and inexpensive means of sensing for robotic platforms compared to systems that employ active sensors such as radar or LIDAR. Using stereo cameras for sensing gives a significant advantage over monocular vision, because the scale of objects can be estimated and there is no need to search of objects in the scale space. In addition, the high resolution 3D depth maps produced by stereo can be used to recognize 3D structures such as ground plane, buildings and vegetation and drastically prune out the location in an image that require additional analysis using appearance classification.

Recent research on pedestrian detection has underscored the significant reduction in the false positives rates (while achieving similar detection rates), when stereopsis is used [1], [2]. These approaches used either sparse stereo for the initial pedestrian detection, resulting in poor foreground/background separation [1], or employed 3D structure from motion to assist in constraining the 2D image based pedestrian detections [2].

This work was supported in part by the Federal Highway Administration under an EARP contract DTFH61-07-H-00039.

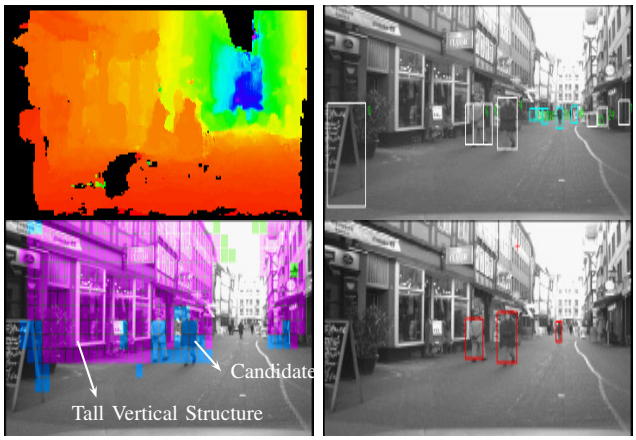


Fig. 1. Overview of our stereo-based pedestrian detection system; Depth-map from stereo, pedestrian boxes from stereo based pedestrian detector, scene labeled by structure classifier and confirmed pedestrian from image based classification.

In contrast, our system makes a more involved use of dense 3D range maps. In order to balance the trade-offs between computational complexity, low false alarms and high detection accuracies, our system, briefly depicted in Fig. 1, implements a number of novel techniques within a layered architecture:(i) at the front-end we use the depth maps to generate initial pedestrian detections using template matching with a 3D human shape and to classify 3D structure elements such as poles, buildings, vegetation to efficiently guide focus of attention. For classification, we learn individual classifiers for individual classes (e.g., trees, poles, buildings) by computing 3D invariant features that are combined within a Markov model framework to enforce the 3D spatial relationships; (ii) Multiple appearance-based pedestrian classifiers are trained separately for three depth ranges to prune the detections generated by stereo. A combination of template matching with 2D human shape contour fragments for localization, along with the standard HoG descriptors for appearance is effectively used to guide focus of attention for computational efficiency while maintaining accuracy; (iii) We use tracking and ego motion estimation to deal with missed pedestrian detections and to remove spurious detections.

Our pedestrian detection system runs at more than 10 Hz on a standard multi-core PC mounted on a moving platform. The system has been tested extensively in a variety of

conditions, as well as on publicly available datasets to allow comparison with other approaches.

The paper is organized as follows. In Section II we discuss several approaches presented in the literature for pedestrian detection. In Section III we give an overview of the whole system proposed. The stereo based pedestrian detection and structure classification are discussed in Section IV and Section V. The image based pedestrian classification is presented in Section VI. In Section VIII we discuss the results obtained using our proposed pedestrian detection system.

II. Related Work

Most research on pedestrian detection has used monocular vision [3], [4], [5], [6], [7] , stereo vision [8], [9], [10], [11], [1] and LIDAR sensing [12]. An overview of several approaches for pedestrian detection can be found in [13].

One of the most popular recent appearance based pedestrian detection algorithms is the Histogram of Oriented Gradients (HoG) method of Dalal and Triggs [3]. They characterized pedestrian regions in an image using HoG descriptors, which are a variant of the well-known SIFT descriptor [14]. Unlike SIFT, which is sparse, the HoG descriptor offers a denser representation of an image region by tessellating it into cells which are further grouped into overlapping blocks.

Another leading real-time, monocular vision system for pedestrian detection in cars was proposed by Shashua et al. [4]. The authors used a focus of attention mechanism to detect window candidates very rapidly. The window candidates (approximately 70 per frame on average) are classified into pedestrian or non-pedestrians using a two stage classifier.

Ess et al. [9], [10], [11] describe a stereo based system for 3D dynamic scene analysis from a moving vehicle, which integrates sparse 3D structure estimation with multi-cue image based descriptors (*shape context* computed at Harris-Laplace and DoG features [15]) to detect pedestrians. The authors show that the use of sparse 3D structure significantly improves of the performance of the pedestrian detector. Still, the best performance cited is 40% probability of detection at 1.65 false positives per image frame. While the structure estimation is done in real time, the pedestrian detections is significantly slower.

Gavrila and Munder [1] propose *PROTECTOR*, a real-time stereo system for pedestrian detection and tracking. PROTECTOR employs sparse stereo to generate putative pedestrian ROIs, which are subsequently pruned using shape (contour) and texture information. The choice of sparse/dense stereo processing stages is justified based on real-time limitations in stereo computation for the whole image. Temporal information is also employed to increase the reliability of the system and to mitigate missing detections, albeit at the price of increased latency of alerting the driver.

Bajracharya et al. [8] describe a real-time stereo-based system that can detect people up to 40 m in lightly cluttered environments. The stereo range maps are projected into a polar-perspective map that is segmented to produce clusters

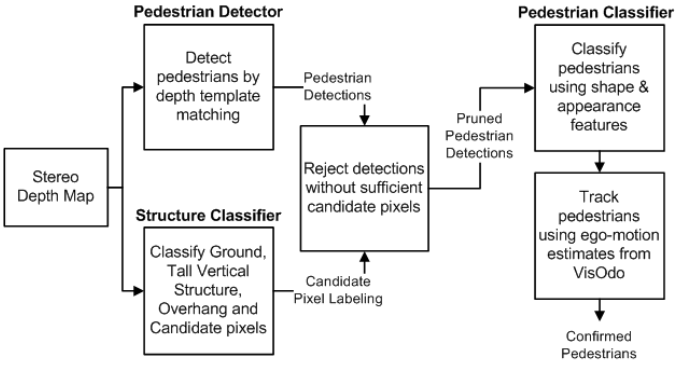


Fig. 2. System architecture of our pedestrian detection system.

of pixels corresponding to upright objects. Geometric features are computed for the resulted 3D point cloud and used to train pedestrian classifiers. Appearance based features are not used for classification.

III. Overall Approach

The proposed approach actively utilizes depth information obtained from stereo computation. Given a depth map of a scene, first a 3D template-based object detector is applied to find candidate target object hypotheses. Simultaneously, the depth map is processed with generic scene descriptor to identify image regions that match predefined structure classes. The scene labeling from these image regions is then combined with object detector hypothesis to produce a final set of object candidates. The resulting hypotheses are passed, first, to an appearance-based pedestrian classifier and then, to a tracker for further processing.

The data from a calibrated stereo-rig is processed at 30 fps on a GPU card to compute dense disparity maps at multiple resolution scales, using a pyramid image representation [16] and a SAD-based stereo matching algorithm. The disparities are generated at three different pyramid resolutions - D_i, $i = 0, \ldots, 2$, with D_0 being the resolution of the input image. In the following, we will refer to disparity and depth images interchangeably.

As shown in Fig. 2, the *Pedestrian Detector* (PD) module takes the individual disparity maps D_i, $i = 0, \ldots, 2$ and converts each individual one into a depth representation. These three depth images are used separately to detect pedestrians using template matching of a 3D human shape model, as shown in detail in Section IV. The *Structure Classification* (SC) module employs a combined $D_0 + D_1 + D_2$ depth map to classify image regions into several broad categories such as tall vertical structure, overhanging structure, ground and candidate regions. The detected pedestrian ROI list from the PD module is pruned by testing overlap with non-candidate region labels from the SC module. ROIs with more than 75% overlap are removed. Next, the *Pedestrian Classification* (PC) module takes in the remaining list of pedestrian ROIs and confirms valid detections by using a cascade of classifiers. Finally, the *Pedestrian Tracking* (PT) module compensates for camera ego-motion using a Visual Odometry module

and performs data-association between classified boxes in successive frames.

IV. Stereo-based Pedestrian Detection

We employ a stereo-based generic object detection algorithm similar to [17] to generate initial pedestrian candidate windows exclusively from range maps. The ground plane (XZ in our convention) is divided into a discrete regular 2D grid. Each specific grid location represents a 3D ground plane location w.r.t the camera, and a pedestrian at this position can be described by a single 3D geometric description of its shape. A pedestrian-shaped cuboid rendered at the given location gives us a depth-template representation of the pedestrian as seen from the camera viewpoint. The depth-templates for all grid locations are pre-computed for efficient matching at run-time.

At run-time, for any given frame, we have the three disparity maps D_i, $i = 0, \ldots, 2$ available. For each 3D template, we look at its distance range Z and then select one of the levels D_i to perform 3D template matching. The level is selected to ensure that, at each location, only the relevant resolution disparity map with sufficient details is used. We use D_2 from $0 - 5\,\text{m}$, D_1 from $5 - 15\,\text{m}$ and D_0 beyond $15\,\text{m}$. For the matching step, the template is correlated with the appropriate level D_i by searching around the X and Z directions, and around the Y (vertical) direction to account for local pitch uncertainty due to calibration errors and bumps in the road surface. The output of this template matching is a correlation score map (over the horizontal 2D grid) from which peaks are selected by non-maximal suppression as in [17]. Around each peak, the area of the correlation score map with values within 60% of the peak score is projected into the image to get the initial pedestrian ROI candidate set.

Note that this detection stage must ensure very small pedestrian miss rates, hence a larger number of peaks obtained by non-maximal suppression is acceptable. We rely on additional steps detailed next to prune these candidates. The initial pedestrian ROI candidate set is pruned, first, by considering the overlap between multiple ROIs: detections with more than 70% overlap with existing detections are removed. After this pruning step, a Canny edge map is computed for each initial pedestrian ROI. The edge pixels are filtered using the disparity map to remove potential background edges by considering a disparity range around the disparity of the detected ROI. A vertical projection of the binary mask corresponding to the remaining edges results in a 1D histogram from which peaks are detected using mean-shift [18]. Each such peak potentially corresponds to a pedestrian due to the high density of edges on people. A new pedestrian ROI is initialized at each detected peak, which is refined first horizontally, and then vertically to get a more centered and tightly fitting bounding box on the pedestrian. The refinement process involves using vertical and horizontal projections, respectively, of binarized disparity maps (similar to using the edge pixels above) followed by detection of peak and valley locations in the computed projections. After these refinements, any resulting overlapping detections are again removed from the detection list. With this approach, we can detect pedestrians upto a range of 40 m.

TABLE I

Structure classes used for scene labeling

V	Tall vertical structure (magenta)
O	Overhanging structure (green)
G	Ground (yellow)
C	Candidate objects (blue)

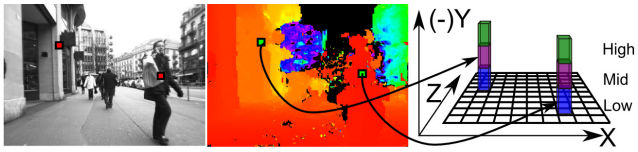


Fig. 3. Vertical Support Histogram. Points from the range map are projected to the bins of a 2D histogram in the ground-plane coordinate system. Each histogram bin captures a different height band. The diagram illustrates a 3-bin histogram.

V. Structure Classification

A key step in our method for pedestrian detection is depth-based classification of the scene into a few major structural components. Given an image and a sparse and noisy range map, the goal is to probabilistically label each pixel as belonging to one of the following scene classes (see Table I for a legend): ground, tall vertical structure, overhang and (pedestrian) object candidates. We provide a brief outline of our structure classification approach below. For details, the reader is referred to [19].

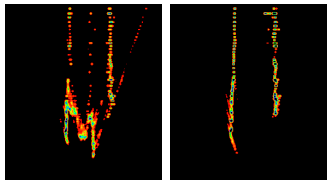
To robustly handle depth-map errors, first, we define a structure called the Vertical Support Histogram to accumulate 3D information over voxels in the vertical direction with respect to a ground plane coordinate system.[1] The ground plane (XZ in our convention) is divided into a regular grid and at each grid cell, a histogram of height-distribution of 3D points is created. All the image pixels that map into a given X, Z coordinate participate in that cell's histogram. The heights, Y coordinate, of all the points in a cell are mapped into a k-bin histogram where each bin represents a vertical height range. We call this structure by the name *Vertical Support Histogram (VSH)* and denote it by V. Fig. 3 shows how image points and the corresponding depth estimates are mapped to 3D distributions for an example histogram with $k = 3$ bins.

Next, for each image pixel, we use its 3D X and Z coordinates to associate the histogram entry $V(X, Z)$ with this pixel. This transfers the 3D representation V to the 2D image and gives us a k-dimensional feature vector for each pixel.

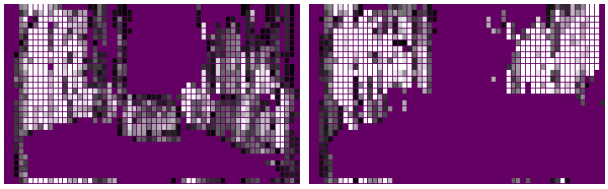
In order to associate each pixel with structural labels, we compute likelihoods for the feature vectors computed above, conditioned on the specific structural labels defined earlier. The likelihood densities are learned by kernel density

[1]The ground plane can be estimated using any of a number of well-known techniques applied to the reconstructed stereo points, see e.g. [2].

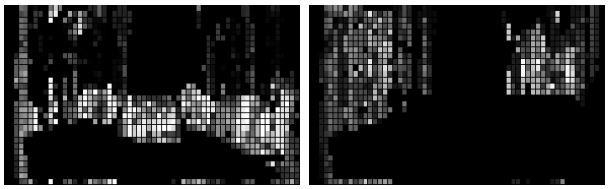
(a) Original frame



(b) Plan views of two VSH bins: $0-2\,\mathrm{m}$ and $2-4\,\mathrm{m}$



(c) The 3D VSH bins projected into the image



(d) Image likelihoods conditioned on the labels: candidate objects and vertical structures, respectively

(e) Labeled structures: tall vertical (magenta) and candidate objects (blue)

Fig. 4. Structure Classification illustrated with a two-bin VSH. In (b), tall vertical objects (like buildings) span both histogram bins while objects with low profile (like vehicles and people) span just the first bin. In (c), the bins are projected back to image space and the structures that projected to the histogram bins can be identified. The bin values provide a feature representation at each pixel from which the label likelihoods in (d) are estimated. These likelihoods can be seen to map well to the actual structure in the scene.

estimation on the feature space defined by a small set of labeled training data. Fig. 4 illustrates the labeling process through a 2-bin VSH.

Next, we make use of the scene-context constraints arising from the camera viewpoint by formulating the labeling problem as an MRF where the smoothness constraints allow us to reason about the relative positioning of the 3D structure labels in the image. Thus, for each pixel, we will consider its neighboring pixel (pixel with which it is 4-connected and is close in its world depth) and define the cost of associating a pair of labels with the two pixels. In our implementation, we define this cost as a binary function which incurs a penalty 1 if the two pixels cannot be neighbors for the given labeling (e.g. a candidate pixel and a overhanging pixel) and 0 otherwise. The MRF was implemented using the max-product belief propagation algorithm [20].

VI. Pedestrian Classification Layer

An image-based classifier is adopted in the system to further evaluate pedestrian hypothesis candidates returned from depth template-based detector. The proposed method combines Histogram of Oriented Gradients (HoG) with contour segments of body parts to address the issues in HoG computation.

In the general object detection schemes based on regression such as Adaboost and SVM [3], [7], [21], it is often required to search for optimal ROI size and position to obtain valid classification scores. This is due to the sensitivity of the classifiers to ROI alignment as rigid placement of the local sampling windows inside ROI become susceptible to different object configuration changes. This results in the need for exhaustive search over multiple positions and scales for each input ROI. False negatives can often occur when people appear against complex background of high textureness because image gradient-based features become fragile in the presence of multiple gradient directions in a local image patch.

These issues are addressed by our approach of combining contours with HoG by (1) local parts alignment and (2) background filtering.

A. Contour+HoG-based Classifier

In this approach, basically a set of shape contours of pedestrian templates is associated with each local sampling window to guide feature localization and filtering process. Local shape contour set is constructed by sampling binary body part contours from representative silhouette database. To cover pose and shape variation of people, each local sampling window typically contains multiple $(5 \sim 10)$ part templates. Figure 5 displays examples of part templates of

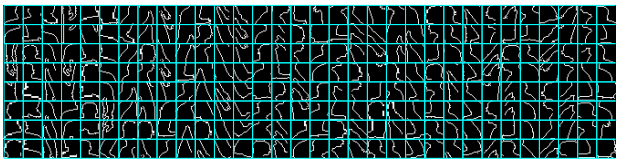


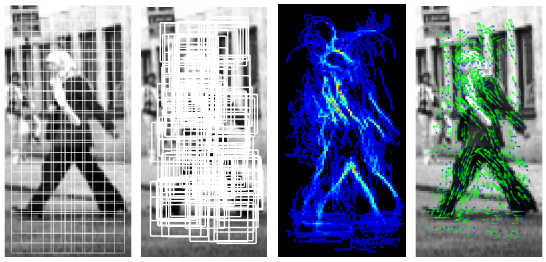
Fig. 5. Example of local contour models. The part vocabulary is sampled from silhouette contour data, where each local vocabulary set can cover body pose and shape variations.

different sampling windows.

The procedure for contour-based matching and evaluation is as follows. First, each local sampling window inside ROI is localized by chamfer matching. Given an input ROI, the chamfer score for each part template is computed in the local region and the associated sampling window is anchored at the maximum correlation based on scores computed from the local template set. The proposed local template-based scheme allows flexibility of matching multiple pose pedestrians effectively without maintaining a large set of global contours and provides a computationally efficient solution.

Secondly, a global foreground mask is composed by overlapping the set of matched local templates, each of which

is weighted by matching scores at the refined position. The foreground mask is used to suppress noisy image features in the background, where in the HoG computation process the gradient values on the matching pedestrian contour parts are enhanced by non-zero weight while the others potentially from the background are filtered out. The procedure of contour matching and filtering is illustrated in figure 6.



(a) (b) (c) (d)

Fig. 6. Procedure of contour+HoG classification. (a) fixed local sampling grid inside ROI, (b) local sampling window refinement from contour matching, (c) composed foreground mask from contour segments, (d) filtered HoG directions underlying masked regions.

Figure 7 shows examples of foreground mask on pedestrians and negative patches. It can be seen that on the positive pedestrian examples, the proposed scheme can refine sampling window position on top of underlying body parts and thus enhance body contours. On the negative image patches however, it produces non-conforming shape and window positions in general.

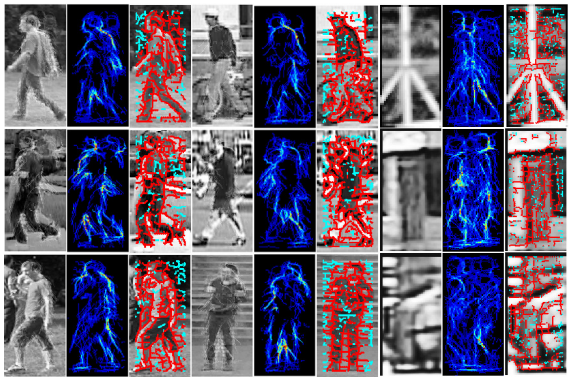


Fig. 7. Foreground mask examples. Each set of three images display original image, foreground mask from matched part templates and the resulting filtered edge map. On the right-most column set is shown the results on negative data. Note that local contour parts can capture global body contours at various poses from its combinations. However, it does not form conforming pedestrian masks on negative patches.

Although this scheme can provide effective solutions to aforementioned issues, it faces a limitation on the low-resolution image patches due to fragility that arises from obscure object boundary in contour extraction and matching steps. To meet this challenge, we adopt a classifier suite that is tuned to different distance ranges. Three classifiers are used where for the closest range of $[0 \sim 20m]$, contour plus HoG classifier is used, while for the distance intervals of $[20 \sim 30m]$ and $[30 \sim 40m]$, a conventional HoG-based classifiers are employed. Each classifier is trained separately with image dataset with each corresponding preset image resolution with Adaboost. The resulting classifiers consist of multiple $(3 \sim 4)$ cascade layers, respectively.

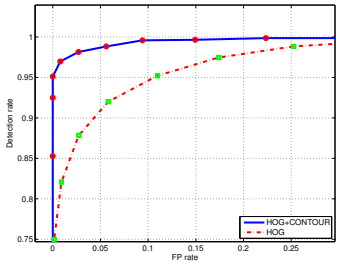


Fig. 8. ROC curve for Contour+HOG v.s. HOG-only classifiers.

Figure 8 shows ROC curve of the Contour+HoG classifier and HOG-only-based classifier that is evaluated on high resolution image dataset. The figure shows distinctive advantage of the Contour+HoG with better performance for both true detection and false positive rate.

VII. Pedestrian Tracking

We adopt simple and effective image correlation-based tracker to recover intermittent missing pedestrians. It consists of 3D feature-based camera motion estimation and image correlation-based tracking.

A. Visual Odometry-based Camera Motion Estimation

We use Visual Odometry (Visodo) [22] to compensate for camera ego-motion. Visodo computes 3D motion of the camera, specifically rotation and translation of a vehicle between adjacent frames with respect to the ground plane. To compute camera motion, Visodo first extracts feature points on each frame, where features are obtained from SIFT corner points. The correspondences between adjacent frames are then established by using RANSAC (random sample consensus)-based point association. Given correspondences, the relative camera motion can be computed by solving the structure from motion equation.

B. Image-based Correlation Tracker

The estimated camera motion parameter is used to predict the location of the detected pedestrian boxes on image (ROI) in the current frame. The Visodo-based prediction is important to accurately locate ROIs under large image motions typically induced by vehicle turning motion, for example.

Given predicted location of ROI from the previous frame $(t-1)$, the new location in the current frame (t) is estimated by patch correlation-based tracker module. The correlation-based tracker refines ROI position by searching through multiple candidate positions and scales in the enlarged prediction window that matches the highest appearance similarity with the corresponding ROI image patch.

TABLE II

EVALUATION DATA USED FOR OUR EXPERIMENTS

Sequence Name	#Frames	Annotations used	
		Ess et al.[9]	Us
Seq01	1000	5193	5122
Seq02	451	2359	1999
Seq03	354	1828	1893

VIII. EXPERIMENTS

A. Data and Evaluation Methodology

Data. We experimentally validate our approach on the publicly available dataset [9]. The dataset consists of four challenging test sequences (640×480 at 15Hz) of busy shopping streets with multiple people moving in different directions, taken on different days and under different weather conditions.

Evaluation Methodology. For evaluation, we use the ground-truth pedestrian annotations available with the dataset. All the sequences are completely annotated up to a distance of $\approx$25 m. However, the annotation set is over-complete with invalid cases such as completely occluded pedestrians for example. So accordingly a subset of annotations is used for evaluation in the previous approaches. The exact subset of annotations used by Ess et al.[9], [10], [8] for their reported results is unfortunately not available thus we composed valid subset based on the following visibility critera. The subset of annotations we use includes pedestrians which: (i) are not significantly clipped by the camera field-of-view, and (ii) are at least 50 pixels high. For a detection to be counted as correct, it has to overlap with an annotation by more than 50% using the intersection-over-union measure [23]. Table II compares the sizes of the annotation subsets used by the current approach with [9] and it is shown to be comparable.

B. Experimental Results

The ROC curves for our system were obtained by varying the decision boundary threshold for the appearance-based classifier stage in our pipeline. Fig. 9(a) shows the performance of different configurations of our system on Seq03. The reported configurations are made by combination of structure classification (SC), appearance classifier (CLS) and tracker (TRK). CLS without SC shows the performance of the appearance classifier directly running on the stereo-based pedestrian detector output. CLS with SC shows the performance when the candidate input ROIs to the appearance classifier are filtered using the labeling from structure classification (SC). In this filtering step, any ROIs with more than 75% non-candidate label patches (from amongst all labeled patches within the ROI) were removed. The remaining boxes were then fed to the appearance-based classifier as before. Note that the performance gain with the use of the SC module clearly indicates the effectiveness of the scheme in pruning out potential false positive cases from structural cues. The curves TRK with and without SC were similarly generated by tracking classifier outputs. TRK with SC corresponds to current end-to-end system configuration at run-time.

Fig. 10. Structure and Appearance Classification. The first row shows the result of structure classification; the second row shows the final classified boxes.

Figs. 9(b)-(d) compare our end-to-end performance (given by the TRK with SC curve) with some approaches in recent literature. The results from [9], [10], [11], [8] are included wherever available.

Unlike other comparable approaches from the literature, our system was not trained on any of the data from this dataset. We used a combination of urban data collected from a moving vehicle along with the public INRIA dataset for training. The proposed system exceeds the reported performance of the leading real-time system [8] and matches with those of the non-real-time approaches [9], [10].

Fig. 10 illustrates examples of cluttered scenes where the structure classification removed significant portion of the scene from consideration by the appearance classifier. Only ROIs detected in the blue regions are processed by the appearance classifier and the confirmed ROIs are shown in red in the bottom row. In Fig. 11 we show some examples of final pedestrian detection output from our system. Note the successful detections at far distances, in complex areas.

Computational Performance. The proposed method is implemented on an in-vehicle pedestrian detection system. Our implementation takes about 25ms per frame for stereo computation, template pedestrian detection and structure classification on an Intel Dual-Core processor. The overall system including the appearance classifier runs at about 10Hz.

IX. CONCLUSION

We have presented a real-time stereo-based pedestrian detection system which incorporates several novel ideas: (i) we exploited dense stereo to classify 3D structures and to detect pedestrians thus reducing the number of false positives that have to be classified using 2D shape and appearance information; (ii) we have used the 2D contour and appearance to improve the classification rates over methods using HoG only; (iii) we have used multiple pedestrian classifiers trained at several depth bands to increase the system performance. The results on publicly available dataset show superior performance than leading approaches in terms of accuracy and speed.

Acknowledgments. The authors thank Ted Camus, Ben

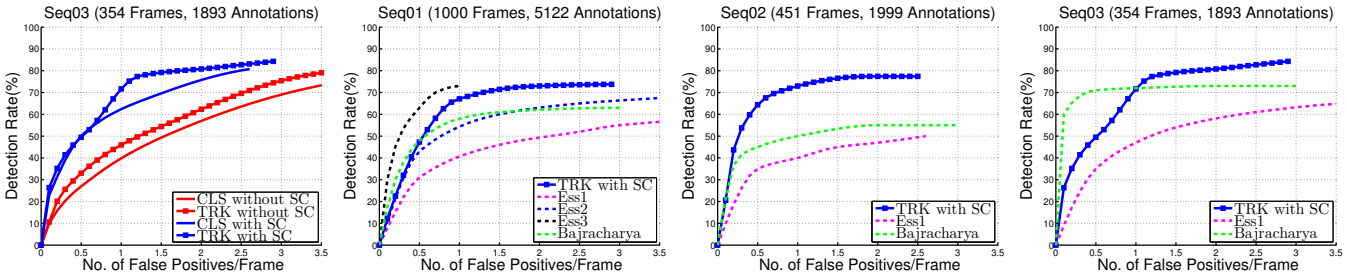


Fig. 9. ROC curves showing our system performance: (a) Improvement in performance using structure classification (SC), appearance classification (CLS) and tracking (TRK), and (b)-(d) comparisons with other representative approaches from the literature: *Ess1* [9], *Ess2* [10], *Ess3* [11], *Bajracharya* [8]. This figure is best seen in color.

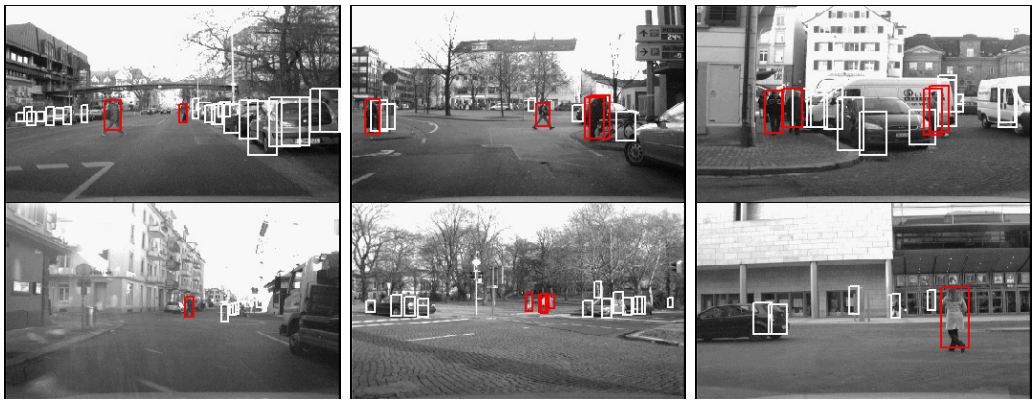


Fig. 11. Examples of pedestrian classification showing coverage at different distances upto 40 m. White boxes denote the output of the stereo-based pedestrian detector after structure classification. Red boxes denote the final pedestrian classifications. The last image shows an example of a failure of our system for the furthest pedestrian.

Southall (Sarnoff), Wei Zhang, Ann Do (FHWA) and John Harding and Jennifer Percer (NHTSA) for helpful discussions and feedback.

REFERENCES

[1] D. M. Gavrila and S. Munder, "Multi-cue pedestrian detection and tracking from a moving vehicle," *IJCV*, vol. 73, pp. 41–59, 2007.

[2] B. Leibe, N. Cornelis, and L. V. G. K. Cornelis, "Dynamic 3d scene analysis from a moving vehicle," in *CVPR*, 2007.

[3] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, 2005, pp. 886–893.

[4] A. Shashua, Y. Gdalyahu, and G. Hayun, "Pedestrian detection for driver assistance systems: Single-frame classification and system level performance," in *In Proc. of the IEEE Intelligent Vehicle Symposium*, 2004.

[5] M. Jones, P. Viola, P. Viola, M. J. Jones, D. Snow, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," in *ICCV*, 2003, pp. 734–741.

[6] P. Sabzmeydani and G. Mori, "Detecting pedestrians by learning shapelet features," in *CVPR*, 2007.

[7] O. Tuzel, F. Porikli, and P. Meer, "Human detection via classification on riemannian manifolds," in *CVPR*, 2007.

[8] M. Bajracharya, B. Moghaddam, A. Howard, S. Brennan, and L. H. Matthies, "Results from a real-time stereo-based pedestrian detection system on a moving vehicle," in *IEEE Workshop on People Detection and Tracking at ICRA*, 2009.

[9] A. Ess, B. Leibe, and L. Van Gool, "Depth and appearance for mobile scene analysis," in *ICCV*, 2007.

[10] A. Ess, B. Leibe, K. Schindler, and L. V. Gool, "A mobile vision system for robust multi-person tracking," in *CVPR*, 2008, pp. 734–741.

[11] A. Ess, B. Leibe, K. Schindler, and L. van Gool, "Moving obstacle detection in highly dynamic scenes," in *In Proceedings of ICRA*, 2009.

[12] K. Fuerstenberg, K. Dietmayer, and V. Willhoeft, "Pedestrian recognition in urban traffic using a vehicle based multilayer laserscanner," in *IEEE Intelligent Vehicle Symposium*, vol. 1, 2002.

[13] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: A benchmark," in *CVPR*, 2009.

[14] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *IJCV*, vol. 60, pp. 91–110, 2004.

[15] K. Mikolajczyk and C. Schmid, "A performance evaluationof local descriptors," *PAMI*, vol. 27, 2005.

[16] P. Burt and T. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. on Communications*, vol. 9, pp. 532–540, 1983.

[17] P. Chang, D. Hirvonen, T. Camus, and B. Southall, "Stereo-based object detection, classification, and quantitative evaluation with automotive applications," in *IEEE International Workshop on Machine Vision for Intelligent Vehicles*, San Diego, 2005.

[18] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *PAMI*, vol. 24, pp. 603–619, 2002.

[19] M. Bansal, B. Matei, H. Sawhney, S. Jung, and J. Eledath, "Pedestrian detection with depth-guided structure labeling," in *IEEE Workshop on Search in 3D and Video (S3DV) at ICCV*, 2009.

[20] Y. Weiss and W. T. Freeman, "On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs," *IEEE Trans. Information Theory, Special Issue on Codes on Graphs and Iterative Algorithms*, vol. 47, 2001.

[21] P. Viola, M. J. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *IJCV*, vol. 63, pp. 153–161, 2005.

[22] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in *CVPR*, 2004.

[23] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2008 (VOC2008)," http://www.pascal-network.org/challenges/VOC/voc2008/workshop/index.html.